What is a load balancer, and how does it decide where traffic goes?

20 November 2026 · 4 min read · blogs.iictp.org/what-is-a-load-balancer

A load balancer sits in front of a group of servers and spreads incoming requests across them. It lets you add capacity by adding servers, and it keeps a service running when one server fails, because it stops sending traffic to servers that are unhealthy.

TL;DR The short version

- A load balancer spreads incoming requests across a group of servers, adding capacity and routing around servers that fail health checks.
- Layer 4 forwards whole connections by address and port; layer 7 understands requests such as HTTP paths.
- Common methods are round robin, least connections and hashing; avoid relying on sticky sessions and make the load balancer itself redundant.

Read online, listen to it, or change the text size: https://blogs.iictp.org/what-is-a-load-balancer

What is the difference between layer 4 and layer 7?

A layer 4 load balancer works with network connections: it sees addresses and ports and forwards whole connections. A layer 7 load balancer understands the application protocol, usually HTTP, so it can route by path, host name or header, terminate TLS and rewrite requests. Layer 7 is more flexible; layer 4 does less work per request.

How does it choose a server?

Common methods are round robin, which takes servers in turn; least connections, which picks the server with the fewest open connections; and hashing on something such as the client address, which sends the same client to the same server. Least connections copes better than round robin when requests vary a lot in cost.

What are health checks for?

The load balancer regularly calls each server, often on a dedicated health endpoint, and removes servers that fail. A good health check tests what the server needs to do useful work, but not every downstream dependency, or one failing database can take every server out at once.

What are sticky sessions, and should you use them?

Sticky sessions send a user back to the same server, usually so session data held in memory is found again. They make scaling and server replacement harder. Keeping session state in a shared store, or in a signed token, lets any server handle any request.

What can go wrong?

The load balancer itself can become a single point of failure, so run it in a redundant pair or use a managed service. Watch for connection draining during deployments, so in-flight requests finish before a server is removed, and for timeouts that differ between the load balancer and the servers behind it.

> Related credential: Cloud Architecture Practitioner. Explore the exam facts and syllabus.

Sources and further reading

1. Kleppmann, M. Designing Data-Intensive Applications. O’Reilly, 2017.
2. Beyer, B. et al. (eds.) Site Reliability Engineering. O’Reilly, 2016. Chapters on load balancing.

Published by IICTP, an independent technical credential body, on 20 November 2026. The current version of this post is at https://blogs.iictp.org/what-is-a-load-balancer. More guides: blogs.iictp.org.